35

引用文献5

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
27 December 2002 (27.12.2002)

PCT

(10) International Publication Number
WO 02/103495 A1

(51) International Patent Classification[7]: G06F 1/00, 9/445

(21) International Application Number: PCT/FI02/00517

(22) International Filing Date: 14 June 2002 (14.06.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
20011278 15 June 2001 (15.06.2001) FI

(71) Applicant *(for all designated States except US)*: **NOKIA CORPORATION** [FI/FI]; Keilalahdentie 4, FIN-02150 Espoo (FI).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **SORMUNEN, Toni** [FI/FI]; Artturintie 6, FIN-33880 Lempäälä (FI). **RÖNKKÄ, Risto** [FI/FI]; Aarikkalankatu 2 as. 4, FIN-33530 Tampere (FI). **KIIVERI, Antti** [FI/FI]; Peikontie 1 F 72, FIN-90550 Oulu (FI).

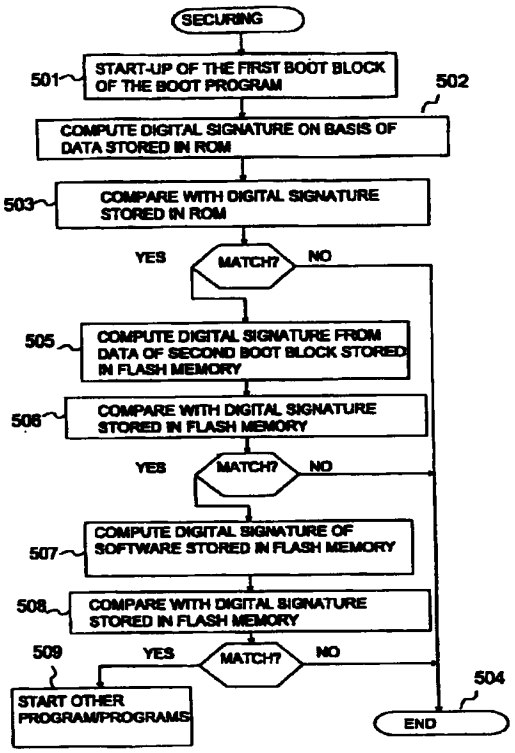
(74) Agent: **TAMPEREEN PATENTTITOIMISTO OY**; Hermiankatu 12 B, FIN-33720 Tampere (FI).

(81) Designated States *(national)*: AE, AG, AL, AM, AT (utility model), AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ (utility model), CZ, DE (utility model), DE, DK (utility model), DK, DM, DZ, EC, EE (utility model), EE, ES, FI (utility model), FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK (utility model), SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW),

[Continued on next page]

(54) Title: A METHOD FOR SECURING AN ELECTRONIC DEVICE, A SECURITY SYSTEM AND AN ELECTRONIC DEVICE



(57) Abstract: The invention relates to a method for securing the trustworthiness of an electronic device (1). At least first (DID, S1, PK1) and second (S2, PK2) check-up data are stored in the electronic device (1). In the method, a boot program is started (501), in which boot program at least first (P1) and second (P2) boot steps are taken. In the first boot step, the trustworthiness of said at least first check-up data (DID, S1, PK1) is examined, wherein if the check-up shows that said at least first check-up data (DID, S1, PK1) is trusted, said second check-up data (S2, PK2) related to at least the second boot step is examined to confirm the trustworthiness of the second boot step. If the check-up shows that at least one second check-up data (S2, PK2) related to the second boot step is trusted, said second boot step (P2) is taken after said first boot step (P1).

WO 02/103495 A1

Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Published:
— *with international search report*

A method for securing an electronic device, a security system and an electronic device

The present invention relates to a method for securing an electronic device as presented in the preamble of the appended claim 1. The invention also relates to a system according to the preamble of the appended claim 16. The invention further relates to an electronic device according to the preamble of the appended claim 28. The invention also relates to a program according to the preamble of the appended claim 31, as well as a storage means according to the preamble of the appended claim 32.

A variety of electronic devices apply programmable control means, such as microprocessors, microcontrollers, programmable logics, and/or application-specific programmable integrated circuits. Such electronic devices contain stored software consisting of one or more programs containing *e.g.* program commands required for the operation of the electronic device. In the storage of such software, a memory is used, of which at least a part is a non-volatile memory, *i.e.* the content of the memory is retained even if the operating voltage of the memory is cut off. Such memories include for example a read-only memory (ROM), a programmable ROM (PROM) and an electrically erasable PROM (EEPROM). At least a part of the memory is normally integrated in the electronic device, but in addition, the memory can be increased in many applications by means of, for example, a memory expansion board. One such memory expansion board is the so-called Flash memory card. The Flash memory is a kind of EEPROM type memory whose content can be changed by electrical programming. The contents of the Flash memory will be retained even after the cutting off of the operating voltages. By means of such an expansion memory, it is easy to provide the electronic device with new software, memory capacity for storing, for example, photographs in a digital camera, for setting access rights *e.g.* in a mobile station, *etc.* The installation of software in an electronic device can also be performed, in a way known as such, by using other storage means, such as a diskette, a CD-ROM, or a DVD.

It is relatively easy to copy software stored on storage means, wherein software providers have developed various methods to prevent the use of copied software. One such method is to use a product ID or the like. Thus, upon starting the program, the user must enter this product ID in the electronic device before the program can be used. However, a problem with such an arrangement is that in connection with copying of the program, the user may have obtained this product ID from the owner of the original software, and also the copied program can then be used. On the other hand, even if the user of the copied software did not know the product ID, the user may try to find out the structure of the program protection, for example by reverse engineering or debugging, wherein the object code of the program is converted to the source code. Thus, the user may succeed in decrypting the copy protection and in modifying the program, for example, in such a way that the copy protection is off, or in such a way that the user resolves the required product ID on the basis of the object code. To make such a possibility more difficult, programs have been developed, in which it is checked at intervals, during the running of the program, that the program has not been tampered with. Thus, the mere decryption of the copy protection upon the booting does not necessarily make it possible to use the copied software for a longer time, unless the user is capable of determining the structure of such copy protection.

It is known to connect a given program unequivocally to a given device in such a way that the program cannot be used in another device. This can be done, for example, by modifying the software on the basis of the hardware-specific serial number or by supplying an installation program which is only functionable in one device on the basis of the hardware-specific serial number. These solutions have the drawback that this protection can be broken up by modifying either the software or the hardware.

To aggravate debugging, an attempt can be made to complicate at least the copy protection part and/or the storage of the product ID in connection with the program code, wherein it becomes more difficult to break up the copy protection. One such solution is presented *e.g.* in the international patent application WO 00/77597.

The patent US 5,131,091 presents a method in which a program stored on a memory card is protected by scrambling the content of the program code with XOR operations. In the scrambling, an encryption bit string stored in a non-volatile memory is used, and finding out the string has been made as difficult as possible. A different encryption bit string is used on memory cards supplied to different users.

A user who has legally acquired the software may also need to secure the origin of the software, because in some cases, a third party may attempt to supply versions modified from original programs and to market them as original programs. Such software may contain, for example, an added virus, or the software is provided with a so-called back door, through which the manufacturer of the modified software may even have access to the local area network of a firm which has installed this modified software. In some cases, the modified software is provided with the property of transmitting, for example, user identifications and passwords entered by the user in the electronic device *e.g.* via a data network such as the Internet to the manufacturer of the modified software, without the user noticing this. To secure the origin of the software, the program can be provided with a digital signature, on the basis of which the user can establish the authenticity of the original software.

In addition to the copy protection of programs, there is also a need to protect other information stored in connection with electronic devices, to prevent misuse. For example, the restriction of access rights to a specific user or specific users is, in connection with some electronic devices, arranged so that the user has a personal smart card, wherein, to use the electronic device, the user inserts the smart card in a card connector provided in the electronic device. As auxiliary authentication, it is also possible to use a user identification, wherein upon turning on of the electronic device, the user must enter this user identification before the electronic device can be used. Such an arrangement is applied *e.g.* in many mobile communication networks, such as the GSM mobile communication network and the UMTS mobile communication network. In a mobile station to be used in such a mobile

communication network, a smart card is inserted, which is called a SIM (Subscriber Identity Module) in the GSM system and a USIM (Universal Subscriber Identity Module) in the UMTS system. In such a smart card, the service provider of the mobile communication network has already set certain subscriber specifications, such as the International Mobile Subscriber Identifier (IMSI). The user identification is also stored in this smart card, wherein the smart card checks the user identification when the mobile station is turned on.

However, the above-presented solutions do not solve the problem that a third party modifies the software in such a way that it can use it itself either in another device or change the operation of the program in this device. Such a problem has come up *e.g.* in connection with mobile stations, in which it has been possible to access the services of a mobile communication network free of charge by making a copy of a mobile station. The software and the international mobile equipment identity (IMEI) of the copied mobile station are identical with those in the original mobile station. A copy is also made of the smart card which is installed in the copied mobile station. Thus, the mobile switching centre does not distinguish between the original mobile station and the copied one.

Yet another drawback in the prior art encryption solutions of software and other data is that if the same encryption key is used for encrypting large quantities of information, the decryption of the encryption key may be successful by analyzing such encrypted information.

With an increase in the data processing capabilities of portable devices, more information can be stored in them, which may also be confidential or otherwise such information that must not be revealed to an outsider. The carrying of portable devices will, however, increase the risk that the portable device is lost or stolen, wherein an attempt must be made to protect the information stored in it with an encryption method. For portable devices, it is normally possible to determine a password which the user must enter in the device at the stage of turning on, until the device can be normally used. However, such a protection is relatively easy to pass, because the passwords used are

normally relatively short, typically having a length of less than ten characters. On the other hand, even if no attempt were made to find out the password, the information contained in the device can be accessed, for example, by transferring the storage means, such as a fixed disk, into another device. If the information contained in the storage means is not in encrypted format, the information stored in the storage means can be easily found out.

It is known that information needed by the user or the device can be encrypted with one key, the encrypted information can be stored in the memory of the device, and it can be decrypted with another key. The key used in asymmetric encryption is different from the key used in decryption. Correspondingly, the key used in symmetric encryption is the same as the key used in decryption. In asymmetric encryption, these keys are normally called a public key and a personal key. The public key is intended for encryption and the personal key is intended for decryption. Although the public key may be commonly known, it can normally not be used to easily determine the personal key corresponding to the public key, wherein it is very difficult for an outsider to find out information encrypted with this public key. One example of a system based on the use of such a public key and a personal key is the PGP system (Pretty Good Privacy), in which the user encrypts the information to be transmitted with the public key of the receiver, and the receiver will then open the encrypted information with his/her personal key. However, there are considerable drawbacks in the systems of prior art. Effective symmetric keys consist of about 100 bits, whereas asymmetric keys consist of about 1000 to 2000 or even up to 4000 bits. If the key string is too short, it is relatively easy to break up with modern data processing equipment which can be called brute force attack. This problem is particularly significant in portable data processing and communicating devices, in which also the limited processing capacity prevents the use of long keys.

It is an aim of the present invention to provide an improved method for securing an electronic device in such a way that a given program is set to function in a given electronic device only. The invention is based on the idea that the boot-up is set to consist of at least two steps in such a

way that in the first step, first check-up data is verified, and if the first check-up data is correct, second check-up data related to the second booting step is verified, wherein if also the second check-up data is correct, it is possible to start the second booting step. More precisely, the method according to the present invention is primarily characterized in what will be presented in the characterizing part of the appended claim 1. The system according to the present invention is primarily characterized in what will be presented in the characterizing part of the appended claim 16. The electronic device according to the present invention is primarily characterized in what will be presented in the characterizing part of the appended claim 28. Further, the software according to the present invention is primarily characterized in what will be presented in the characterizing part of the appended claim 31. Further, the storage means according to the present invention is primarily characterized in what will be presented in the characterizing part of the appended claim 32.

The present invention shows remarkable advantages compared to solutions of prior art. In the electronic device according to the invention, the equipment identity is stored in a memory which is made as difficult as possible to modify. Furthermore, in an advantageous embodiment, the equipment identity is verified with a digital signature, wherein the public key or some key identification information used in the verification is stored in the electronic device. Thus, by checking the digital signature, it is possible to verify, with a high probability, whether the digital signature corresponds to the equipment identity of the electronic device. One equipment identity is set permanently in the device and another is set in the signed data which is called a certificate. Now, by checking the signature, it is possible to find out the authenticity and author of the certificate. It is thus verified that the permanent equipment identity of the device and the equipment identity contained in the certificate are identical. By the method according to the invention, it can be secured that only a given program operates in a specific electronic device. It is thus possible to significantly reduce the economic losses to program providers, caused by the copying of software. It is also possible to improve the position of the users of electronic devices, because, by the solution of the invention, the operation of pirate

electronic devices and software can be made significantly more difficult. Thus, the authorized user will not be charged any costs for the use of such a copied electronic device which corresponds to the user's electronic device. By the method of the invention, the origin of the software can be verified, wherein the user of the software can be relatively sure that the origin of the software corresponds to that indicated, and that the software does not contain any viruses, back doors, or the like. The invention also makes it possible that the software of the electronic device cannot be modified in an unauthorized manner so that it would function after the modifications.

In the electronic device according to the invention, the size of the internal read-only memory of the circuit can be kept relatively small, because the integrity of the programs on the external memory (flash or some other type if memory) can be verified inside the chip. This also makes it possible that a majority of the programs of the electronic device can also be replaced after the manufacture of the electronic device, and also the planning of the programs is easier.

In an advantageous embodiment of the invention, the equipment identity used in the control of the access rights of the programs is independent of the possible IMEI code of the electronic device. Thus, the manufacturer of the device may change the IMEI code, if necessary. Furthermore, the length of the equipment identity can be shorter than the IMEI, wherein upon storing the equipment identity, a smaller quantity of expensive memory capacity will be required than when applying solutions of prior art.

In the following, the invention will be described in more detail with reference to the appended drawings, in which

Fig. 1 shows an electronic device according to a preferred embodiment of the invention in a reduced block chart,

Fig. 2 shows the structure of a boot program in an electronic device applying the method according to an advantageous embodiment of the invention,

Fig. 3 illustrates the manufacturing and delivery of software to an electronic device by a security system according to an advantageous embodiment of the invention,

Fig. 4 illustrates the manufacturing and delivery of software to an electronic device by a security system according to another advantageous embodiment of the invention,

Fig. 5 shows the operation of a boot program according to a preferred embodiment of the invention in a flow chart, and

Fig. 6 shows a known principle on forming a digital signature.

The following is a description on the operation of an electronic device 1 according to an advantageous embodiment of the invention in connection with the method of the invention. The electronic device 1 used can be any electronic device which contains means for running programs. Advantageously, the electronic device 1 preferably comprises operating system software or the like, by which the essential functions of the electronic device are controlled and by which the running of other programs (applications) can be controlled in the electronic device 1. Non-restrictive examples of such electronic devices 1 to be mentioned in this context are a mobile station and a computer.

The electronic device 1 according to an advantageous embodiment of the invention, shown in Fig. 1, comprises a control block 2 containing means 2a for running programs. These means comprise, for example, a micro controller unit MCU and a digital signal processing unit DSP. In addition, the control block 2 preferably comprises an application specific integrated circuit ASIC, in which it is possible to implement, for example, at least part of the logic functions of the electronic device. Furthermore, the control block 2 of the electronic device 1 shown in Fig. 1 is preferably provided with a read-only memory 2d, of which at least a part is a one time programmable ROM (OTPROM) 2e, and a random access memory 2f. However, it is obvious that these memories 2d, 2e, 2f can also be implemented as memories separate from the

control block 2. The electronic device also comprises memory means 3 outside the control block, preferably comprising at least a read-only memory 3a, a programmable read-only memory 3b and a random access memory 3c. At least a part of the read-only memory 3a is implemented in such a way that its content cannot be changed by the user. It is also possible to connect a memory expansion to the electronic device 1 of Fig. 1, by placing a memory expansion block 4 in memory connection means 5. The memory expansion block 4 is, for example, a Flash memory card, but also other memory expansion means can be applied in connection with the invention. Preferably, the electronic device 1 is also provided with a user interface UI which comprises a display 6, a keyboard 7, and audio means 8, such as an earpiece/a speaker and a microphone. The electronic device 1 according to an advantageous embodiment of the invention, shown in Fig. 1, also comprises means 9 for performing mobile station functions, for example a GSM mobile station and/or a UMTS mobile station. Furthermore, the electronic device 1 preferably comprises means 10 for connecting an identity card 11, such as a SIM card and/or a USIM card, to the electronic device 1.

Figure 2 shows the structure of the boot program of the electronic device 1, in which the method according to an advantageous embodiment of the invention is applied. The boot program is divided into at least two boot blocks P1, P2, of which the first boot block P1 performs the initial booting operations of the first step. The second boot block P2 performs further check-ups in a situation in which no errors to prevent the start-up were detected in the first boot block.

The security method according to the present invention, consisting of at least two steps, functions in the following way. The operation is illustrated in the flow chart of Fig. 5. In the start-up of the electronic device 1, the control block 2 starts to run the boot program (block 501 in Fig. 5). This is performed in a way known as such by setting the address register of the control block 2 to a given initial address containing that program command of the boot program which is to be performed first. This program command is located in a first boot block P1. After this, the running of the program is preferably continued by taking

the required steps for initializing the device, which are prior art known by anyone skilled in the art and do not need to be discussed in this context. The first boot block P1 comprises a first check-up step to check first check-up data (first security data). In the first check-up step *e.g.* the device ID or the like stored in the one time programmable ROM 2d will be checked (block 502). This device ID is indicated by the reference DID in Fig. 2. Furthermore, it is possible to check that the program code of the first boot block P1 has not been modified. The checking is preferably performed in the control block 2 by computing a digital signature by using at least said device identity DID and possibly also at least part of the boot program stored in the read-only memory 2d, 2e. In the computing of the digital signature, the same algorithm and the same data are used, by which the digital signature was computed in connection with the manufacture of the electronic device 1 by a secret key of the device manufacturer, as will be presented below in this description. This digital signature is preferably stored in the programmable read-only memory 3b (indicated with reference S1 in Fig. 2), but it is obvious that it can also be stored, for example, in the same read-only memory 2d, 2e in which the device identity DID has been stored. The digital signature can be verified by using the public key PK1 which corresponds to the secret key used in the signature and is stored in the read-only memory 2d, 2e. After the computing of the digital signature, a comparison is made between the digital signature computed in the control block 2 and the digital signature S1 stored in the one time programmable read-only memory 2d, 2e (block 503). If the comparison shows that the digital signatures match, it is possible to continue the booting. In other cases, it is obvious that an attempt has been made to modify the electronic device 1 and/or the identity data DID contained in it and/or the boot program, wherein as a result, the normal operation of the device is prevented, for example by switching off the electronic device (block 504). This part of the boot program which makes the checking is stored in the memory of the electronic device 1 in such a way that it cannot be changed without breaking the electronic device 1. One useful solution is to use the internal, one time programmable read-only memory 2e of the control block 2 for the storage.

When the booting is continued, the next step is to take the second check-up step of the boot program before starting any other programs PG1, PG2, PG3. The program code corresponding to the second check-up step is in the first boot block P1. In the second check-up step, the authenticity of the second boot block P2 of the boot program is checked. The second boot block P2 of the boot program is preferably stored in the electrically erasable programmable read only memory (EEPROM) 3b, such as a Flash memory. A digital signature is computed by using at least part of the boot program stored in the read-only memory 3a, 3b (block 505). Also the digital signature S2 of the second boot block of the boot program is stored in the same memory 3b. The computation of this digital signature S2 applies some data that can be verified, such as a part of the program code of the second boot block of the boot program as well as the secret key of the manufacturer of the electronic device 1. The public key PK2 corresponding to this secret key is also stored in the memory 3b. The computed digital signature is compared with the digital signature stored in the memory 3b (block 506), and if the signatures match, the booting of the electronic device 1 can be continued further. However, if the signatures do not match, the normal operation of the device is prevented, for example by halting the electronic device.

The data to be checked (second check-up data, second security data) in the second check-up step may have been formed, for example, by computing compressed data H, *e.g.* by a hash function, from programs PG1, PG2, PG3, parameters, device identities DID, IMEI, or the like, stored in the programmable read-only memory 3b. This compressed data H is signed with the secret key and stored in the programmable read-only memory 3b. In this case, the checking is performed by verifying the authenticity of this signature.

In an advantageous embodiment of the invention, information is transmitted from the first boot step to the program performing the second boot step, about the location of the check-up program to be used in the second boot step and the public key PK2.

After the above-presented second check-up step has been successful, it is possible to run the second boot block P2 of the boot program stored in the memory 3b. In this second boot block P2 of the boot program, *e.g.* some other data stored in the read-only memory 3b is verified, *e.g.* according to the above-presented principles by computing one or more digital signatures (block 507) and comparing it/them with the corresponding digital signatures stored in the read-only memory (block 508). Such data to be verified include, for example, device-specific information, such as the device identity DID, the international mobile equipment identity IMEI, as well as service provider specific information, such as a SIM lock SL, whereby the mobile station can be set to operate with only one or more specific SIM/USIM cards, and/or an operator lock, whereby the electronic device 1 is set to operate with a SIM/USIM card of a specific mobile telephone operator only. The running of the second boot block P2 of the boot program can be continued, if the signatures and other possibly performed verifications were in order. It should be mentioned that in some applications, the above-mentioned device identity DID may be formed on the basis of the international mobile equipment identity IMEI, but they may also be independent of each other.

Next, at least another check-up step is to be taken to examine the still unverified part of the program code of the programs PG1, PG2, PG3, or at least some of them (block 509). In this check-up step, it is possible to apply the above-presented principles, wherein at least another digital signature and the information required for its verification are stored in the memory 3b.

After all the verifications determined for starting the electronic device 1 have been performed, it is possible to boot other programs PG1, PG2, PG3 (block 510), after which the electronic device 1 can be used normally.

The above-presented public keys can also be verified with a digital signature (= certificate) to obtain greater certainty of the origin of the public keys. In this case, not only the public keys but also the correspond-

ing digital signatures are stored in the memory 2d, 2e, 3b and verified before they are used for other check-up measures.

Although, in the above description, the boot program was only divided in the first P1 and second P2 boot blocks, it is obvious that in connection with the invention, the boot program can also be divided into more than two boot blocks P1, P2. Thus, each boot block involves verification of at least the next boot block, before the operation moves on to the next boot block. In the verification, information is used, of which at least a part is stored in this boot block next in the order.

The above-described verifications can also be made after the booting, during the normal operation of the electronic device 1. The aim of this is to prevent, for example, the replacement of the smart card, after the booting, with a smart card whose use in said electronic device 1 is unauthorized, or the replacement of the external memory 3b with a memory containing a modified program code.

At the stage of manufacturing of the electronic device 1 according to the invention, and/or at the stage of updating the software, the required check-up data and programs are formed in the memory 2d, 2e, 3a, 3b preferably in the following way. The program codes required in the verifications are stored in the control block 2, including the first boot block P1 of the boot program, the program for computing the digital signature, and the encryption and decryption algorithm/algorithms. This step is represented by block 301 in Fig. 3. The manufacturer also stores at least a part of the device identity DID in the one time programmable memory 2e of the control block (block 303). Furthermore, the public key PK1 of the manufacturer and the digital signature S1, required for the verification of the first boot block P1 and the device identity, are stored in the one time programmable memory 2e. After performing the necessary storage in the one time programmable memory 2e, this one time programmable memory 2e is set, if necessary, in a state in which no more changes can be made in the memory. The aim of this is to prevent the changing of, *e.g.* single bits in the device identity DID or in another part of the one time programmable read-only memory 2e. At the stage of assembling the components (block 302),

also the control block 2 containing the one time programmable memory 2e is installed in the circuit board of the electronic device (not shown). The manufacturer stores the other blocks P2 of the boot program and possible application programs *e.g.* in the programmable memory 3b and/or in the one time programmable memory 3a (blocks 304 and 305). Also the public key PK2 of the manufacturer used for checking the second boot block P2 of the boot program, the digital signature S2 as well as a possible certificate are stored in the memory 3b. After this, the electronic device 1 can be delivered to a dealer or a service provider, such as a mobile telephone operator. Thus, when a purchaser of the electronic device 1 enters a subscriber contract with the service provider, the electronic device 1 can be set to function with one or more smart cards 11, such as a SIM card, or any smart card of the service provider. Thus, the service provider or the seller sets a SIM lock SL or the like as well as the device identity IMEI in the electronic device 1. If necessary, a certificate is retrieved from a certificate data base CDB, to be used for verification of the authenticity of the data in connection with the booting of the electronic device 1 in a way described above in this description. These definitions are stored in the memory 3, preferably in the programmable memory 3b. After this, the electronic device 1 is ready for use. It is obvious that the operations provided by the dealer/service provider above can also be performed by the device manufacturer or a service company authorized by the device manufacturer. Thus, the data about the service provider and the purchaser are transmitted to the enterprise at which the data are stored.

On the basis of the above-described confirmation data, the service provider can make sure that the electronic device 1 according to the invention, being connected to the services of the service provider, really is the device whose identity code is stored in the electronic device 1. Furthermore, it is guaranteed that the electronic device 1 and the data contained in it have not been subjected to unauthorized modification.

The invention can also be applied to update software and other data in the electronic device 1. This can be implemented, for example, by the

dealer and/or a service company, *e.g.* with the arrangement shown in Fig. 4. For example, the user wants to have a new version of the operating system to be installed in the electronic device 1. The new operating system version has been supplied by the provider of the operating system to said enterprise AS, or it is downloaded via a data network from the provider of the operating system. In connection with the downloading, the device identity DID is preferably given, and possibly also the identity of the service provider (block 401 in Fig. 4). When downloading the operating system, the necessary verifications are made that the receiver is really authorized to the downloading and that the user is authorized to receive the new version of the operating system in his/her electronic device (block 402). The operating system to be downloaded can now be provided with the data about the device identity DID, the public key of the program provider, and/or a digital signature (arrow 403), wherein the running of the operating system version can be limited to said electronic device 1 only. Thus, no other copy protection will be necessary. The new operating system version is transmitted to the electronic device 1 (arrow 404), to be stored in the programmable memory 3b by a method known as such (block 405).

There are a number of encryption methods known which can be applied in connection with the present invention. Symmetric encyption methods to be mentioned in this context include Data Encryption Standard (DES), Advanced Encryption Standard (AES), and Rivest's Cipher 2 (RC2). One asymmetric encryption method is Rivest, Shamir, Adleman (RSA). Also so-called hybrid systems have been developed, employing both asymmetric encryption and symmetric encryption. In such systems, asymmetric encryption is normally used when an encryption key to be used in symmetric encryption is transmitted to the receiver, wherein the symmetric encryption key is used in the encryption of actual information.

For the transmission of public keys to be used in asymmetric encryption, a system has been developed which is called Public Key Infrastructure (PKI). This system comprises servers in which the public keys are stored and from which a user needing a key can retrieve the key. Such a system is particularly applicable for use by companies, wherein

the company itself does not need to transmit its public key to anyone who wishes to transmit information to the company in an encrypted manner.

For digital signatures, several systems have been used, such as the RSA, Digital Signatures Algorithm (DSA), and Elliptic Curve Cryptography (ECC). In connection with these systems, algorithms which compress the information to be signed are used, including Secure Hash Algorithm (SHA-1) and Message Digest 5 (MD5) to be mentioned in this context. Figure 6 shows the forming of a digital signature in a principle view. The data 601 to be signed is led to a block 602 performing a hash function (compressing function). After this, the compressed data formed by the hash function is signed 603 with a secret key SK. The signature 604 is connected to the data 601 to be signed. At the stage of verifying the signed data, the data confirmed with the signature is led to a block 605 performing the hash function, for producing a hash code 606. The signature is verified 607 by using a public key PK corresponding to the signatory's secret key, after which the hash code 606 is compared 608 with the data formed in the verification 607 of the signature. If the data match, the signed data can be relied on with a high probability.

The steps according to the invention can be largely implemented with program commands of the software running means 2a in the control block 2 of the electronic device 1.

The invention can also be applied, for example, in the implementation of language versions of software related to the electronic device 1. Thus, for each language version, a set of programs is formed, containing the desired language definitions. The device identity is set as data in this set of programs, wherein the programs can only be used in a given device. To secure this, the mechanisms complying with the present inventions are applied. On the other hand, the solution of the invention can also be applied in such a way that it is possible to determine, in the electronic device 1, which language version or versions can be installed and run in said electronic device 1.

It is obvious that the present invention is not limited solely to the above-presented embodiments, but it can be modified within the scope of the appended claims.

Claims:

1. A method for securing the trustworthiness of an electronic device (1), in which electronic device (1) at least first (DID, S1, PK1) and second (S2, PK2) check-up data are stored, in which method the start-up (501) of a boot program is performed, **characterized** in that in the boot program, at least first (P1) and second (P2) boot steps are taken, that in the first boot step, the trustworthiness of at least said first check-up data (DID, S1, PK1) is examined, wherein if the check-up shows that said at least first check-up data (DID, S1, PK1) is trusted, at least said second check-up data (S2, PK2) related to the boot step is examined to confirm the trustworthiness of the second boot step, wherein if the check-up shows that said at least one second check-up data (S2, PK2) related to the second boot step is reliable, said second start-up step (P2) is taken after said first boot step (P1).

2. The method according to claim 1, **characterized** in that in the forming of said first check-up data (DID, S1, PK1), program commands of said first boot step are used at least partly.

3. The method according to claim 1 or 2, **characterized** in that in the forming of said first check-up data (DID, S1, PK1), at least part of the device identity (DID) of the electronic device is used, which is stored in the electronic device (1).

4. The method according to claim 3, **characterized** in that if said device identity (DID) of the electronic device does not match with the device identity used in the formation of the first check-up data (DID, S1, PK1), the normal operation of the electronic device (1) is prevented.

5. The method according to any of the claims 1 to 4, **characterized** in that said first check-up data (DID, S1, PK1) is formed by a digital signature (S1) by using a secret key, that information relating to the public key (PK1) corresponding to the secret key is stored in the electronic device (1), and that information relating to said public key (PK1) stored in the electronic device (1) is used in the confirmation of said first check-up data (DID, S1, PK1).

6. The method according to claim 5, **characterized** in that information relating to the public key (PK1) is stored in a one time programmable read-only memory (2e), and that said first check-up data (DID, S1, PK1) is stored in an external memory of the electronic device (1).

7. The method according to any of the claims 1 to 6, **characterized** in that in the formation of said second check-up data (S2, PK2), program commands of said second boot step are used at least partly.

8. The method according to any of the claims 1 to 7, **characterized** in that at least said first check-up data (DID, S1, PK1) is stored in a read-only memory (2e).

9. The method according to any of the claims 1 to 8, **characterized** in that at least a part of the program commands of said first boot step (P1) is stored in a read-only memory (2e).

10. The method according to any of the claims 1 to 9, **characterized** in that the program commands of said first boot step (P1) and the program commands of said second boot step (P2) are stored in different memories (2d, 2e, 3).

11. The method according to claim 9 or 10, **characterized** in that said second check-up data (S2, PK2) are stored in the same memory, in which the program commands of said second boot step (P2) are stored.

12. The method according to any of the claims 1 to 11, **characterized** in that said second check-up data (S2, PK2) is formed by a digital signature (S2) by using a secret key, that the public key (PK2) corresponding to the secret key is stored in the electronic device (1), and that said public key (PK2) stored in the electronic device (1) is used in the confirmation of said second check-up data (S2, PK2).

13. The method according to claim 12, **characterized** in that for forming the digital signature (S2), a set of data is selected, the data of

the selected set is compressed for forming a compression (H), and the digital signature (S2) is formed on the basis of said compression (H).

14. The method according to any of the claims 1 to 13, **characterized** in that in the electronic device (1), at least one program (PG1, PG2, PG3) is run, whose start-up is performed in the second boot step (P2) and which at least one program (PG1, PG2, PG3) is provided with at least third check-up data, that before starting said program (PG1, PG2, PG3), said third check-up data is examined to secure the trustworthiness of said program (PG1, PG2, PG3), wherein if the check-up of the trustworthiness of said program (PG1, PG2, PG3) shows that said program (PG1, PG2, PG3) is reliable, at least one said program (PG1, PG2, PG3) is started.

15. The method according to any of the claims 1 to 14, **characterized** in that in said boot program stored in a read-only memory the trustworthiness of the first boot steps (P1) are checked before performing said first boot steps (P1).

16. A system for securing the trustworthiness of an electronic device (1), in which electronic device (1) at least first (DID, S1, PK1) and second (S2, PK2) check-up data are stored, and the electronic device (1) comprises means (2) for starting a boot program, **characterized** in that the system comprises means (2) for running the boot program in at least first (P1) and second (P2) boot steps, means for examining the trustworthiness of at least said first check-up data (DID, S1, PK1) in said first boot step (P1), and means for examining said second check-up data (S2, PK2) related to at least a second start-up step to confirm the trustworthiness of the second boot step, wherein if said at least first check-up data (DID, S1, PK1) and said at least one second check-up data (S2, PK2) related to the second boot step are reliable on the basis of said check-ups, said second boot step (P2) is arranged to be performed after said first boot step (P1).

17. The system according to claim 16, **characterized** in that in the formation of said first check-up data (DID, S1, PK1), program commands of said first boot step are used at least partly.

18. The system according to claim 16 or 17, **characterized** in that in the forming of said first check-up data (DID, S1, PK1), the device identity (DID) of the electronic device is used, which is stored in the electronic device (1).

19. The system according to claim 18, **characterized** in that it comprises means for comparing the equivalence of the device identity (DID) of the electronic device stored in said electronic device (1) and the device identity used in the formation of the first check-up data (DID, S1, PK1), and means for stopping the boot program, if said device identity (DID) of the electronic device does not correspond to said device identity used in the formation of the check-up data (DID, S1, PK1).

20. The system according to any of the claims 16 to 19, **characterized** in that it comprises means for forming said first check-up data (DID, S1, PK1) with the digital signature (S1) by using a secret key, means for storing information relating to the public key (PK1) corresponding to the secret key in the electronic device (1), and means for using the public key (PK1) stored in said electronic device (1) in the confirmation of said first check-up data (DID, S1, PK1).

21. The system according to any of the claims 16 to 20, **characterized** in that in the formation of said second check-up data (S2, PK2), program commands of said second boot step are used at least partly.

22. The system according to any of the claims 16 to 21, **characterized** in that it comprises a one time programmable read-only memory (2e), in which at least said first check-up data (DID, S1, PK1) is stored.

23. The system according to claim 22, **characterized** in that at least a part of the program commands of said first boot step (P1) is stored in a read-only memory (2e).

24. The system according to claim 23, **characterized** in that said second check-up data (S2, PK2) is stored in the same memory in

which the program commands of said second boot-up step (P2) are stored.

25. The system according to any of the claims 16 to 24, **characterized** in that it comprises means for forming said second check-up data (S2, PK2) with the digital signature (S2) by using a secret key, means for storing the public key (PK1) corresponding to the secret key in the electronic device (1), and means for using the public key (PK2) stored in said electronic device (1) in the confirmation of said second check-up data (S2, PK2).

26. The system according to any of the claims 16 to 25, **characterized** in that it comprises means (2) for running a program in an electronic device (1), which program is provided with at least third check-up data, means for examining said third check-up data to confirm the trustworthiness of said program, means for starting said program in said second start-up step (P2), if said check-up of the trustworthiness of the program showed that said program is reliable.

27. The system according to any of the claims 16 to 26, **characterized** in that it comprises a program loading system (AS), in which programs are stored for loading, means (9) for transmitting first check-up data (DID, S1, PK1) from an electronic device (1) to the program loading system (AS), means (402, CDB) for confirming the first check-up data (DID, S1, PK1), means (402) for adding the first check-up data (DID, S1, PK1) to a program to be loaded in the electronic device (1), and means (404, AS) for transmitting the program to the electronic device (1).

28. An electronic device (1) comprising means for securing the trustworthiness of an electronic device (1), in which electronic device (1) at least first (DID, S1, PK1) and second (S2, PK2) check-up data are stored, and the electronic device (1) also comprises means (2) for starting a boot program, **characterized** in that the electronic device (1) comprises means (2) for running the boot program in at least first (P1) and second (P2) boot steps, means for examining the trustworthiness of at least said first check-up data (DID, S1, PK1) in

said first boot step (P1), and means for examining said second check-up data (S2, PK2) related to at least a second boot step to confirm the trustworthiness of the second boot step, wherein if said at least first check-up data (DID, S1, PK1) and said at least one second check-up data (S2, PK2) related to the second boot step are reliable on the basis of said check-ups, said second boot step (P2) is arranged to be performed after said first boot step (P1).

29. An electronic device according to claim 28, **characterized** in that it comprises means (9) for performing mobile station functions.

30. An electronic device according to claim 29, **characterized** in that it comprises means (9) for downloading programs via a mobile station network.

31. A program for securing the trustworthiness of an electronic device (1), in which electronic device (1) at least first (DID, S1, PK1) and second (S2, PK2) check-up data are stored, and which program includes program commands for performing the start-up of a boot program (501), **characterized** in that the program also comprises program commands for performing at least first (P1) and second (P2) boot steps in the boot program, program commands for examining the trustworthiness of at least said first check-up data (DID, S1, PK1) in the first boot step, program commands for examining at least said second check-up data (S2, PK2) related to the second start-up step to secure the trustworthiness of the second boot step, program commands for performing said second boot step (P2) after said first boot step (P1) if said at least first check-up data (DID, S1, PK1) and said at least one second the check-up data (S2, PK2) related to the second boot step are reliable on the basis of said check-ups.

32. A storage means (2d, 2e) for storing a program used for securing the trustworthiness of an electronic device (1), in which electronic device (1) at least first (DID, S1, PK1) and second (S2, PK2) check-up data are stored, and which program includes program commands for performing the start-up of a boot program (501), **characterized** in that the program stored in the storage means also comprises program

commands for performing at least first (P1) and second (P2) boot steps in the boot program, program commands for examining the trustworthiness of at least said first check-up data (DID, S1, PK1) in the first boot step, program commands for examining at least said second check-up data (S2, PK2) related to the second boot step to secure the trustworthiness of the second boot step, program commands for performing said second start-up step (P2) after said first boot step (P1) if said at least first check-up data (DID, S1, PK1) and said at least one second check-up data (S2, PK2) related to the second boot step are reliable on the basis of said check-ups.

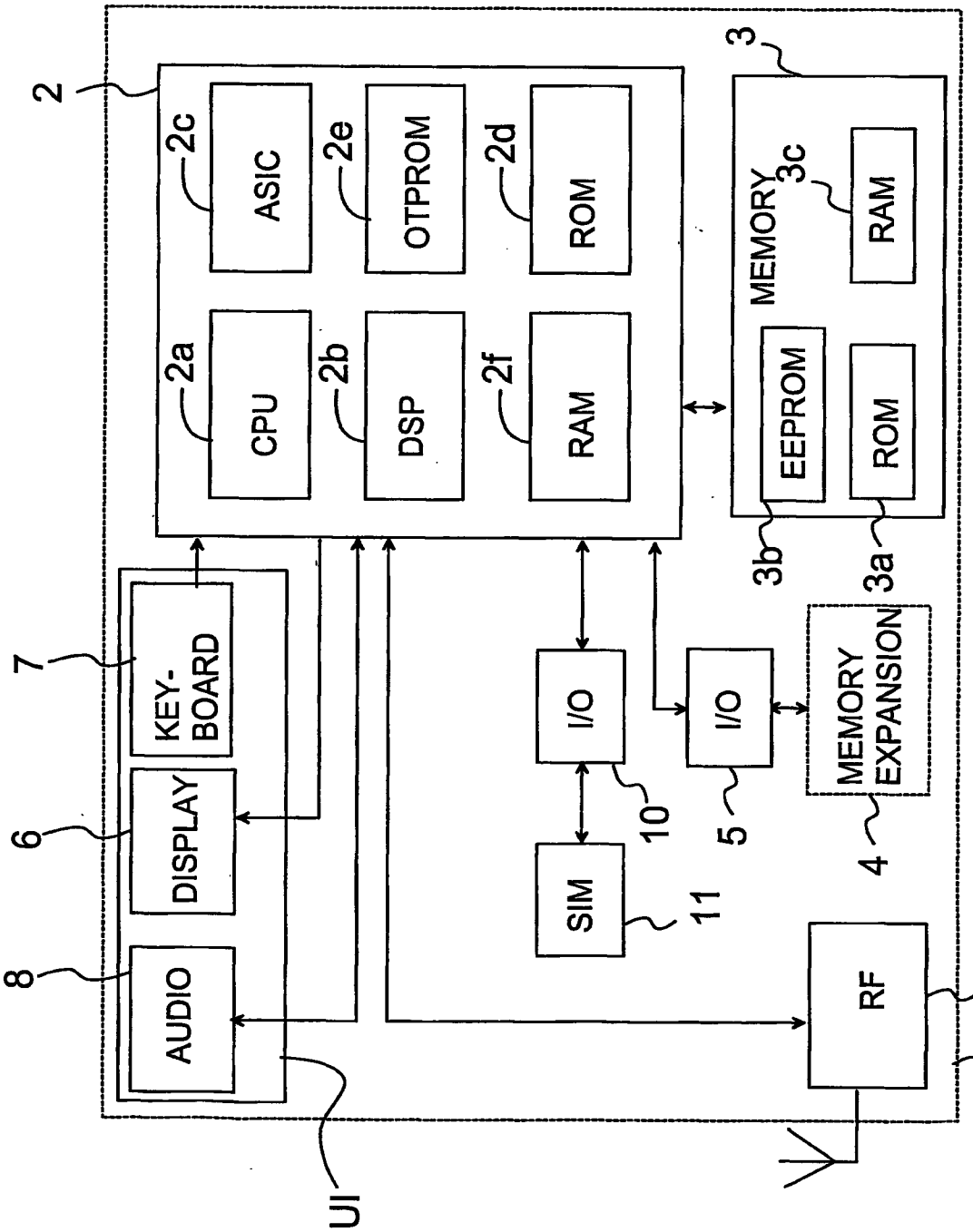

2
2c
ASIC
2e
OTPROM
2d
ROM
2a
CPU
2b
DSP
2f
RAM
3
MEMORY
3c
RAM
EEPROM
ROM
3b
3a
7
KEY-BOARD
6
DISPLAY
8
AUDIO
UI
I/O
10
SIM
11
I/O
5
MEMORY EXPANSION
4
RF
9
1

Fig. 1

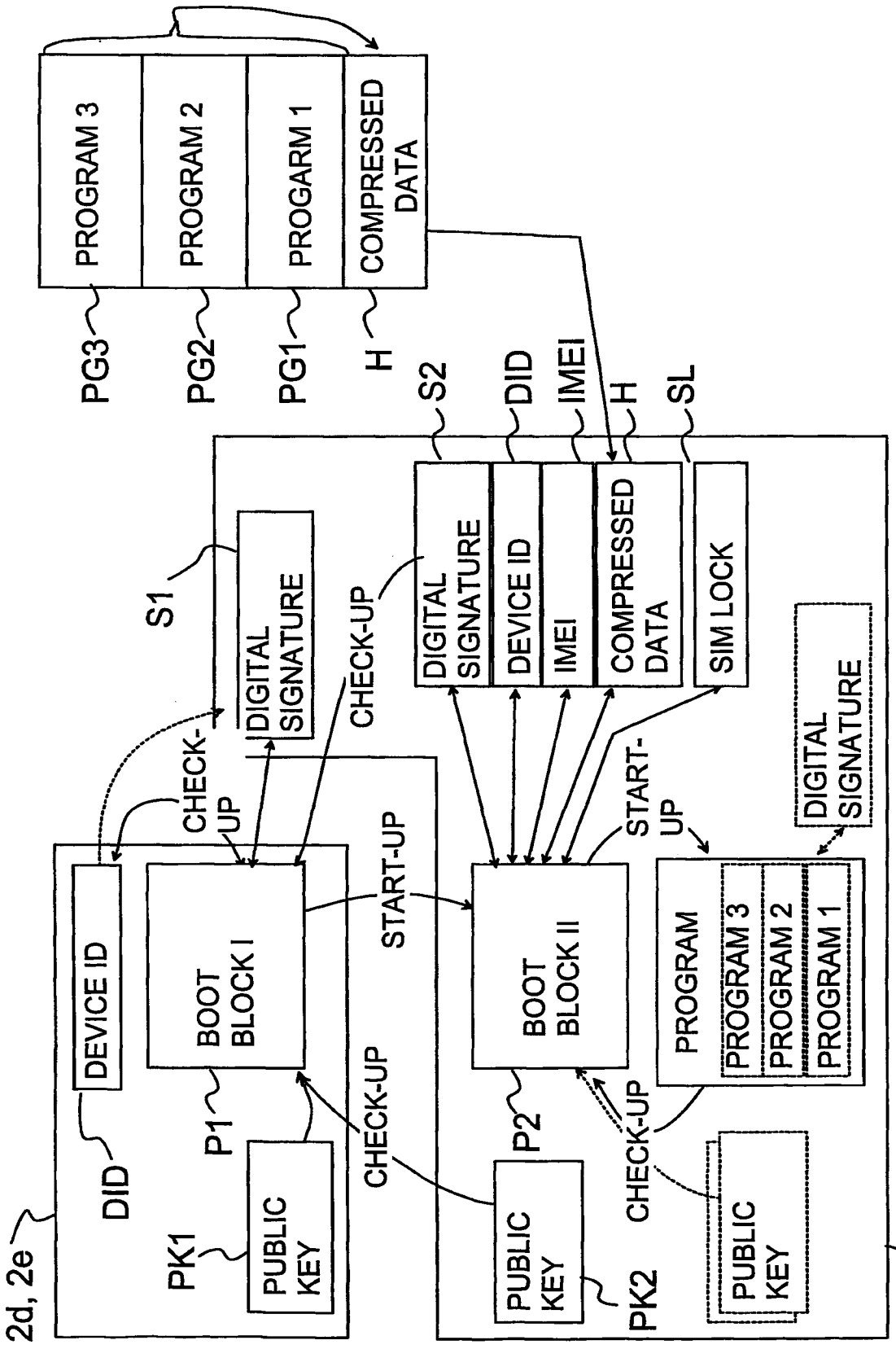

PROGRAM 3
PROGRAM 2
PROGARM 1
COMPRESSED DATA
PG3
PG2
PG1
H
S2
DID
IMEI
H
SL
S1
DIGITAL SIGNATURE
DEVICE ID
IMEI
COMPRESSED DATA
SIM LOCK
CHECK-UP
CHECK-UP
START-UP
START-UP
DIGITAL SIGNATURE
DEVICE ID
BOOT BLOCK I
BOOT BLOCK II
PROGRAM
PROGRAM 3
PROGRAM 2
PROGRAM 1
DID
P1
P2
CHECK-UP
CHECK-UP
PK1
PUBLIC KEY
PUBLIC KEY
PUBLIC KEY
PK2
2d, 2e
3b

Fig. 2

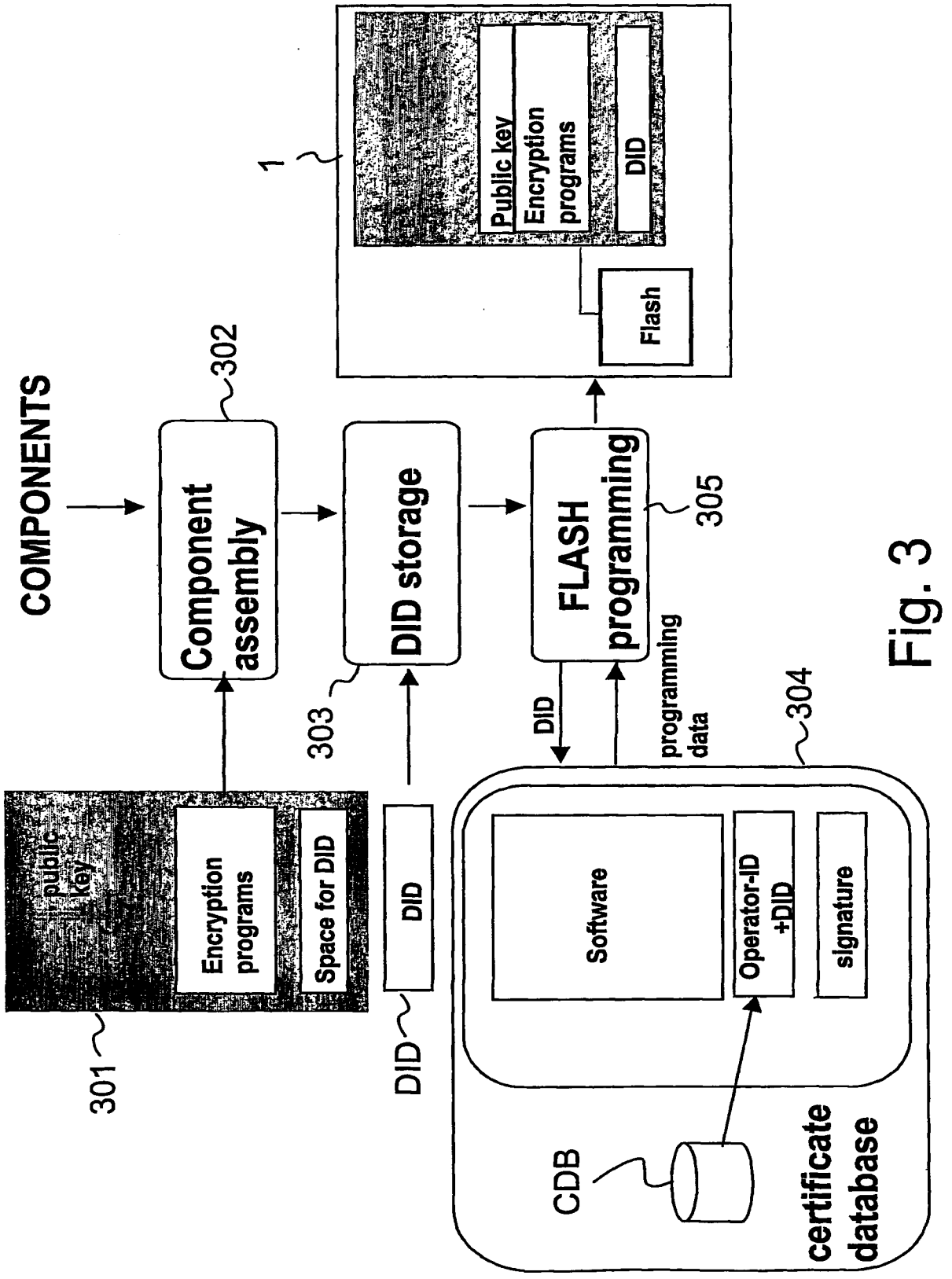

COMPONENTS
301
public key
Encryption programs
Space for DID
302
Component assembly
303
DID storage
DID
1
Public key
Encryption programs
DID
Flash
FLASH programming
305
DID
programming data
304
Software
Operator-ID +DID
signature
CDB
certificate database

Fig. 3

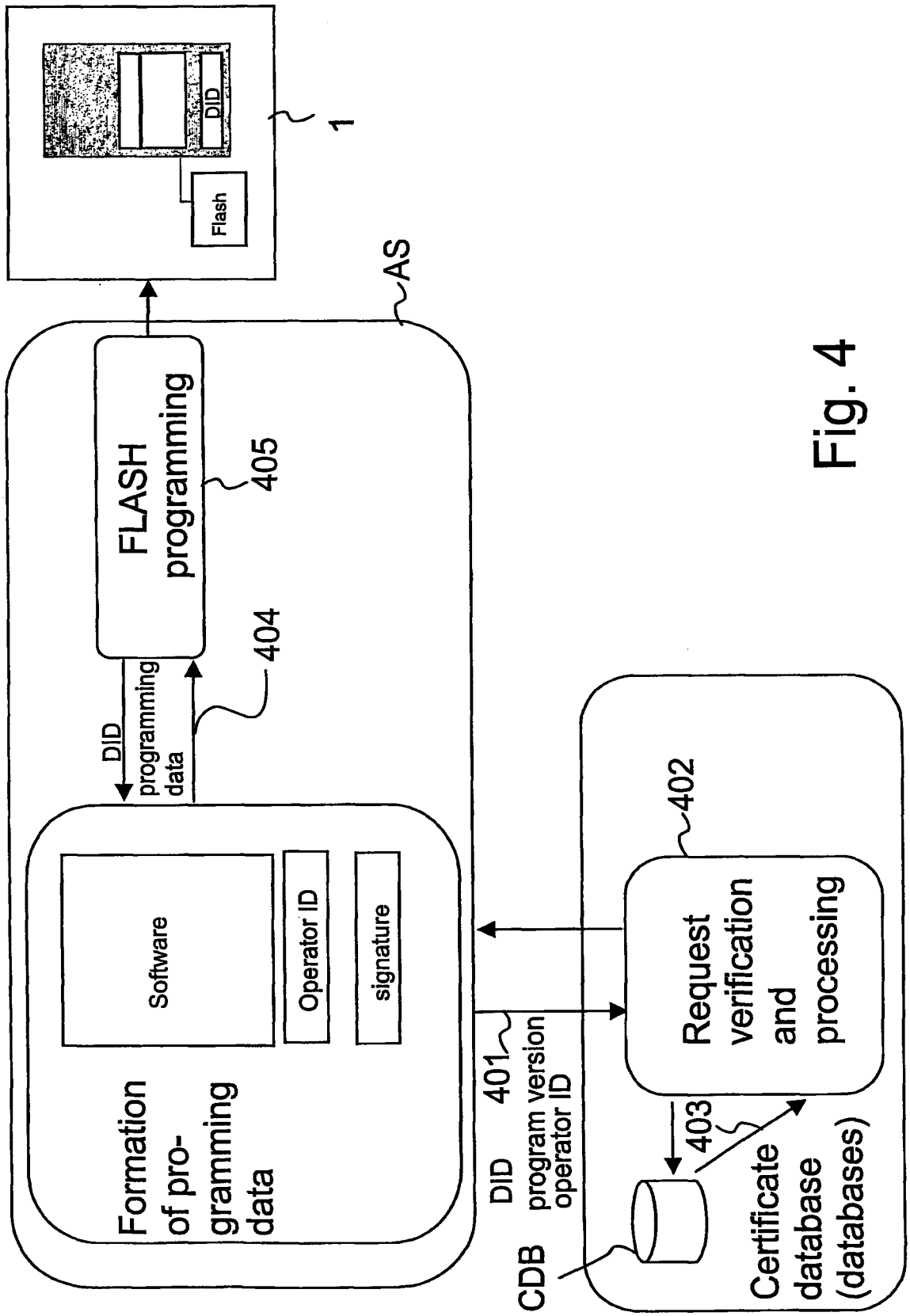

DID
Flash
1
AS
FLASH programming
405
404
DID
programming data
Software
Operator ID
signature
Formation of pro-gramming data
401
DID
program version
operator ID
CDB
402
Request verification and processing
403
Certificate database (databases)

Fig. 4

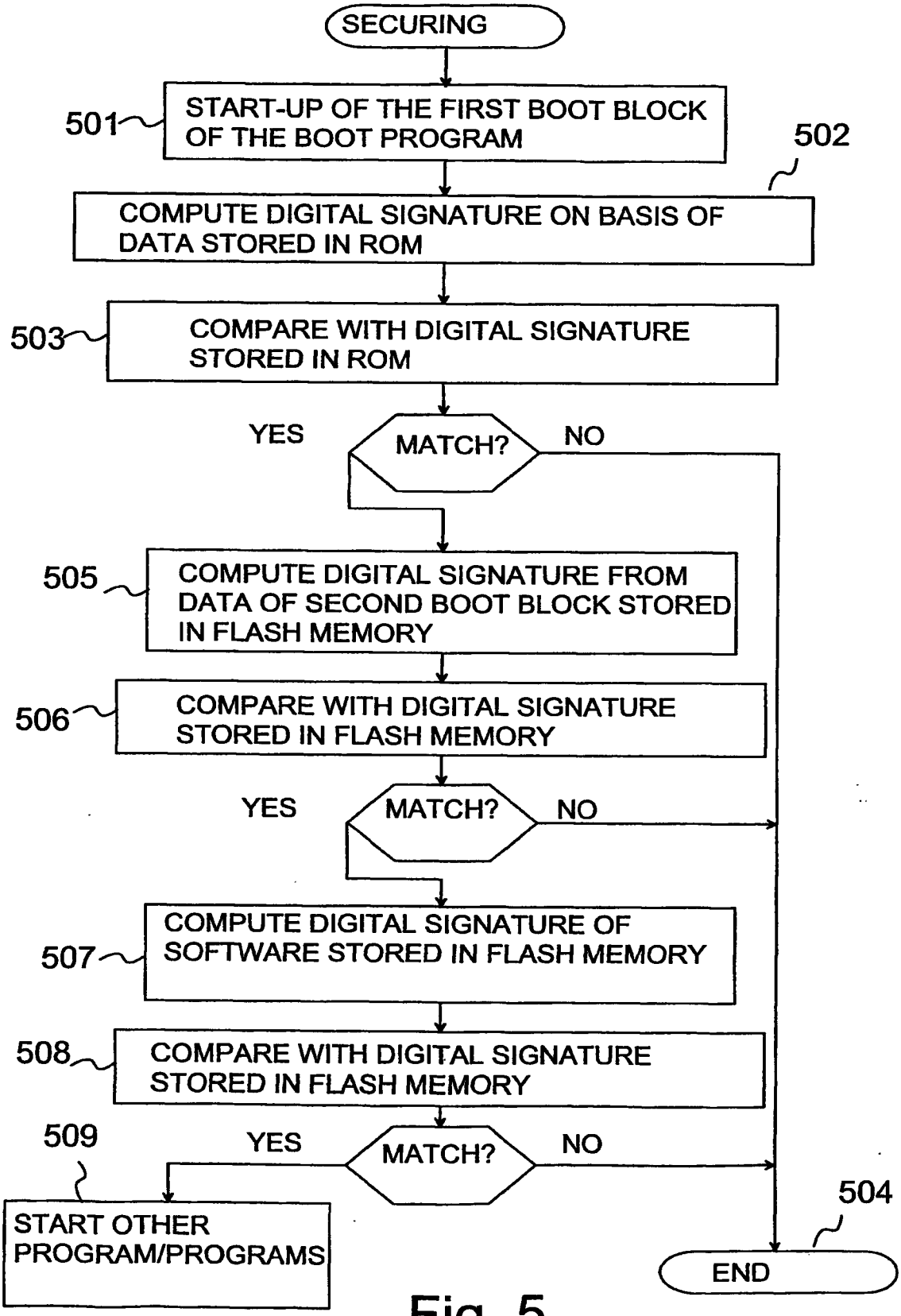

SECURING
501
START-UP OF THE FIRST BOOT BLOCK OF THE BOOT PROGRAM
502
COMPUTE DIGITAL SIGNATURE ON BASIS OF DATA STORED IN ROM
503
COMPARE WITH DIGITAL SIGNATURE STORED IN ROM
YES
MATCH?
NO
505
COMPUTE DIGITAL SIGNATURE FROM DATA OF SECOND BOOT BLOCK STORED IN FLASH MEMORY
506
COMPARE WITH DIGITAL SIGNATURE STORED IN FLASH MEMORY
YES
MATCH?
NO
507
COMPUTE DIGITAL SIGNATURE OF SOFTWARE STORED IN FLASH MEMORY
508
COMPARE WITH DIGITAL SIGNATURE STORED IN FLASH MEMORY
509
YES
MATCH?
NO
START OTHER PROGRAM/PROGRAMS
504
END

Fig. 5

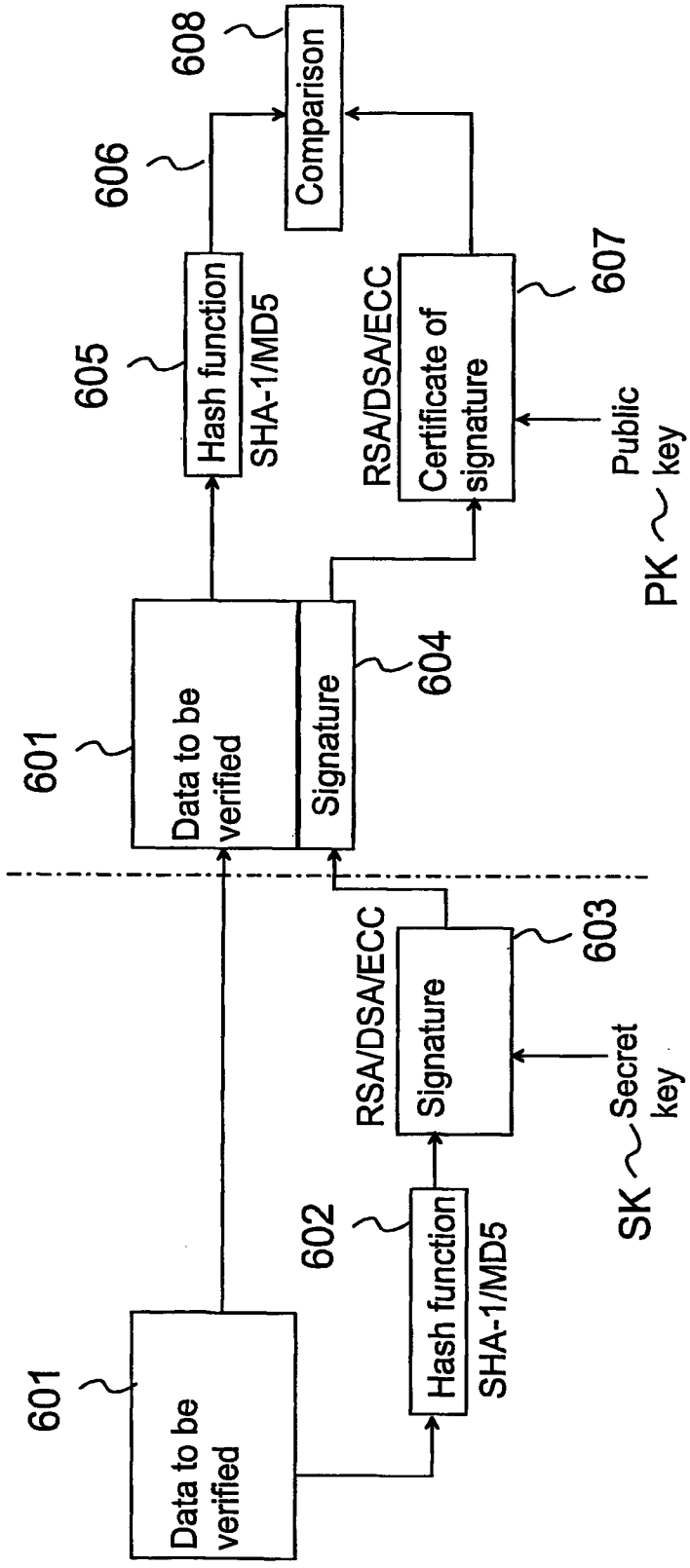

608
Comparison
606
605
Hash function
SHA-1/MD5
RSA/DSA/ECC
Certificate of
signature
607
Public
key
PK
601
Data to be
verified
Signature
604
RSA/DSA/ECC
Signature
603
Secret
key
SK
602
Hash function
SHA-1/MD5
601
Data to be
verified

Fig. 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI 02/00517

A. CLASSIFICATION OF SUBJECT MATTER

IPC7: G06F 1/00, G06F 9/445
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC7: G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

SE,DK,FI,NO classes as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-INTERNAL, WPI DATA, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5802592 A (CHESS, D.M. ET AL.), 1 Sept 1998 (01.09.98), claims 1-28	1-32
X	US 5919257 A (TROSTLE, J.), 6 July 1999 (06.07.99), column 5, line 27 - line 46	1-32
X	EP 0606771 02 (INTERNATIONAL BUSINESS MACHINES CORP), 20 July 1994 (20.07.94), claims 1-9, abstract	1-4,7-11, 14-19,21-24, 26-32
Y		5,6,12,13, 20,25

[X] Further documents are listed in the continuation of Box C. [X] See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance: the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance: the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
25 Sept 2002	26 -09- 2002
Name and mailing address of the ISA/ Swedish Patent Office Box 5055, S-102 42 STOCKHOLM Facsimile No. +46 8 666 02 86	Authorized officer Kristoffer Ogebjer/LR Telephone No. +46 8 782 25 00

Form PCT/ISA/210 (second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI 02/00517

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 0656587 A1 (INTERNATIONAL BUSINESS MACHINES CORP), 7 June 1995 (07.06.95), claims 1-8	1-4,7-11, 14-19,21-24, 26-32
Y		5,6,12,13, 20,25
	--	
Y	US 6032257 A (OLARIG, S.P. ET AL.), 29 February 2000 (29.02.00), claims 1,4	5,6,12,13, 20,25
	--	
A	EP 0816970 A2 (SUN MICROSYSTEMS, INC), 7 January 1998 (07.01.98), claims 3,4	1-32
	--	
A	EP 1076279 A1 (HEWLETT-PACKARD CO), 14 February 2001 (14.02.01), claim 1	1-32
	--	

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/FI 02/00517

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5802592 A	01/09/98	NONE	
US 5919257 A	06/07/99	NONE	
EP 0606771 02	20/07/94	JP 6236280 A US 5379342 A	23/08/94 03/01/95
EP 0656587 A1	07/06/95	US 5509120 A	16/04/96
US 6032257 A	29/02/00	NONE	
EP 0816970 A2	07/01/98	US 6138236 A	24/10/00
EP 1076279 A1	14/02/01	EP 1203278 A EP 1204910 A WO 0113198 A WO 0113199 A	08/05/02 15/05/02 22/02/01 22/02/01